A Constraint to Automatically Regulate Document-Length Normalisation

Ronan Cummins
Discipline of Information Technology
National University of Ireland, Galway
ronan.cummins@nuigalway.ie

Colm O'Riordan
Discipline of Information Technology
National University of Ireland, Galway
colmor@it.nuigalway.ie

ABSTRACT

Retrieval functions in information retrieval (IR) are fundamental to the effectiveness of search systems. However, considerable parameter tuning is often needed to increase the effectiveness of the retrieval. Document length normalisation is one such aspect that requires tuning on a per-query and per-collection basis for many retrieval functions.

In this paper, we develop an approach that regularises the level of normalisation to apply on a per-query basis. We formally describe the interaction between query-terms and document length normalisation using a constraint. We then develop a general pre-retrieval approach to adapt a number of state-of-the-art ranking functions so that they adhere to the constraint.

Finally, we empirically demonstrate that the adapted retrieval functions outperform default versions of the original retrieval functions, and perform at least comparably to tuned versions of the original functions, on a number of datasets. Essentially this regulates the normalisation parameter in a number of retrieval functions on a per-query basis in a principled manner.

Categories and Subject Descriptors: H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval: Models

General Terms: Experimentation, Measurement, Performance

Keywords: Retrieval Functions, Constraints

1. INTRODUCTION

Document length normalisation is known to be of crucial importance to the effectiveness of retrieval functions. However, the level of normalisation to apply is known to be both query and collection specific [2, 11]. As a result, considerable parameter tuning needs to be conducted before a retrieval function is close to optimal on a given collection and set of queries. Retrieval functions derived from several models of retrieval [13, 15, 1] use the ratio of the document length to the average document length to normalise a document. This paper deals with these types of retrieval functions. The contribution of this paper is three-fold:

- We formalise a constraint regarding the interaction of query-length and document-length normalisation (Section 3).
- We develop a general approach to adapt retrieval functions so that they adhere to the new constraint (Section 4).
- We empirically demonstrate that the new versions of the retrieval functions are comparable to tuned versions of the original functions (Section 5).

In the next section we discuss background and related-work.

CIKM'12, October 29–November 2, 2012, Maui, HI, USA.
ACM 978-1-4503-1156-4/12/10.

2. BACKGROUND AND RELATED WORK

Typically, a user submits a query Q to an information retrieval (IR) system M. The system, which has an index of N documents, scores each document D according to some scoring, or retrieval, function $S(Q, D)$. The system then returns all documents that contain at least one query-term (the returned set RET) in decreasing order of $S(Q, D)$. Each document and query consists of a set of terms t in the collection C (i.e. $t \in C$). Although all documents are ranked according to a static query, the query has important characteristics that affect retrieval effectiveness. Recent research [14] has outlined that most modern retrieval functions can still be thought of as a inner-product of term-weights from a query and document vector as follows:

$$S(Q, D) = \sum_{t \in Q} G(t, Q, D) \cdot F(t, Q, D) \quad (1)$$

where $G(t, Q, D)$ is a query-side term-weighting function and $F(t, Q, D)$ is a document-side term-weighting function. The axiomatic approach to information retrieval [9, 10] models the document in an inductive manner and describes a retrieval function by modelling the manner in which the score of a document (via $F(t, Q, D)$) changes as on-topic or off-topic terms are added to the document. This approach has provided an interesting and novel way of defining a basic underlying mathematics of retrieval that has been adopted by others [6, 7, 4, 5, 14]. Currently most ranking functions apply a simple weighting function to the terms in the query-vector ($G(t, Q, D)$) and concentrate on deriving high performing term-weights for the document vector ($F(t, Q, D)$). The query-side weighting function (i.e. $G(t, Q, D)$) may have several interesting constraints that have not yet been correctly captured. Recent research [14] has begun to look at the change in the ranking of documents when terms are added to the query.

Research into the automatic tuning of document length normalisation has previously been conducted [11, 12, 3].

Some approaches [11] measure what they call the 'normalisation effect' and hypothesise that this is similar across all collections. The approach described [11] is computationally expensive as all the documents that contain query-terms need to be analysed in order to tune the document length normalisation parameter. Others [3] have incorporated the query-length into the vector space model and conducted experiments on Chinese and English corpora suggesting that the query-length should be incorporated in other existing ranking functions. In this work, we focus on adapting three modern ranking functions, namely, pivoted document length normalisation (Piv) [15], a probabilistically-derived ranking function (BM25) [13], and a retrieval function based on the divergence-from-randomness model (PL2) [1]. The retrieval functions used in this paper are generalised by equation 1. Specifically, we use the versions of the retrieval functions that adhere to many of the original constraints and are described in recent research (Table 1 in [14]).

3. FORMAL CONSTRAINTS

In the original axiomatic work, the retrieval functions were described by the change in document score as on-topic, or off-topic, terms were added to the document. However, we will see that the relative ranking of documents can also change as terms are added to the query (i.e. a reformulated query). In this section we will introduce a new constraint and motivate it accordingly.

3.0.1 QLNC

Let Q be a query and q be a query-term such that $q \in Q$. Assume D_1 and D_2 are two documents such that $q \in \{D_1 \cap D_2\}$. Furthermore, let us assume that $S(Q, D_1) = S(Q, D_2)$ and $|D_2| > |D_1|$. If we reformulate the query by adding a term t where $t \notin \{Q \cup D_1 \cup D_2\}$, then $S(Q \cup t, D_1) > S(Q \cup t, D_2)$.

This normalisation constraint ensures that longer documents get penalised more when terms mismatch. This constraint controls the interaction of document length normalisation with the query length and can be considered a query length normalisation constraint **QLNC**. It ensures that there is greater length normalisation applied to documents for longer queries. The reformulated query contains an extra term that appears in neither D_1 nor D_2. For the reformulated query, the score of both documents should be lower than with the original query Q. However, the score of D_2 should now be lower than D_1 because it is *more* off-topic. In most retrieval functions the score of a document does not decrease when a query-term does not match a document.

From a probabilistic perspective, D_2 has a greater prior probability of matching any new query-term (i.e. t) because it is a longer document. Therefore, it should also be penalised more if t does not occur. This constraint will help to regulate document length normalisation so that it is query-dependent (which previous research would tend to suggest). Longer queries require greater normalisation (i.e. higher b in BM25, higher s in Piv, lower c in PL2). Table 1 outlines the retrieval functions that adhere to **QLNC**. We can see that the only modern retrieval function that adheres to the new constraint is the Dirichlet-Priors Language model (Dir)[1]. Therefore, we only focus on the three aforementioned retrieval functions for the experiments in this paper.

Table 1: Adherence of Retrieval Functions to QLNC

	Piv	BM25	PL2	Dir
QLNC	No	No	No	Yes

4. QUERY-LENGTH NORMALISATION

We will now outline a general method that can be employed to adapt the necessary retrieval functions so that they adhere to **QLNC**. The only retrieval function that adheres to **QLNC** is Dir. Therefore, we will include some feature of the query (one that increases with length) into the document length normalisation components of Piv, BM25, and PL2. The general approach taken is to adapt the normalisation aspect of the retrieval functions so that each document appears shorter (than its true length) when presented with a short query, while making the document appear closer to its true length for a long query.

4.1 Incorporation of a Prior Probability

The document length normalisation used in Piv, BM25, and PL2 consists of the ratio of the document length to the average document length (i.e. $\frac{|D|}{avg\_dl}$). Document length normalisation penalises longer documents as they have a higher prior probability of containing different query-terms. However, for short queries, the probability that a document chosen at random will contain a query-term, is quite low. We hypothesise that for this type of query, the level of normalisation to apply, should be low. Conversely, long queries (that also may contain terms that appear in many documents i.e. high df_t terms), the level of document length normalisation (penalisation) should be higher, as there is a higher prior probability that a randomly selected document will contain a query-term. The probability that a document D chosen at random from the collection contains at least one query-term is given by:

$$P(q \in any\_D) = 1 - \prod_{t \in Q} \left(\frac{N - df_t + 0.5}{N + 1}\right) \qquad (2)$$

where q is any query-term. We can see that this probability increases as new query-terms are added to the query. The values of 0.5 and 1 ensure that the probability strictly increases as the query-length increases and can be interpreted as hyper-parameters used for smoothing. We incorporate this probability into the normalisation aspect of Piv, BM25, and PL2 by multiplying it by the document length ($|D|$) as follows:

$$\frac{|D| \cdot P(q \in any\_D)}{avg\_dl} = \frac{|D|}{|C|} \cdot P(q \in any\_D) \cdot N \qquad (3)$$

The right hand side of the equation is re-written by substituting $avg\_dl = |C|/N$, and shows that the normalisation

[1]To our knowledge the only retrieval function that satisfies this constraint is the Dirichlet-Priors language model which incorporates a term penalisation factor ($log(u/(u + |D|))$ where u is a tuning parameter) into every query-term. Furthermore, it has been shown that Dir is one of the most effective modern retrieval functions [9]. We only mention this retrieval function (Dir) for completeness.

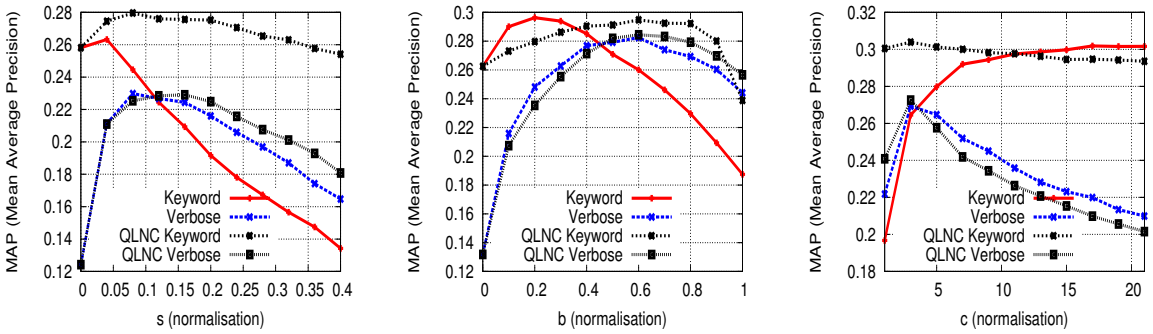

Figure 1: Effectiveness (Mean Average Precision) of Piv, BM25, and PL2 (left to right) vs adapted versions of each Retrieval Function (labelled QLNC) for both Keyword and Verbose queries on WT2G Collection

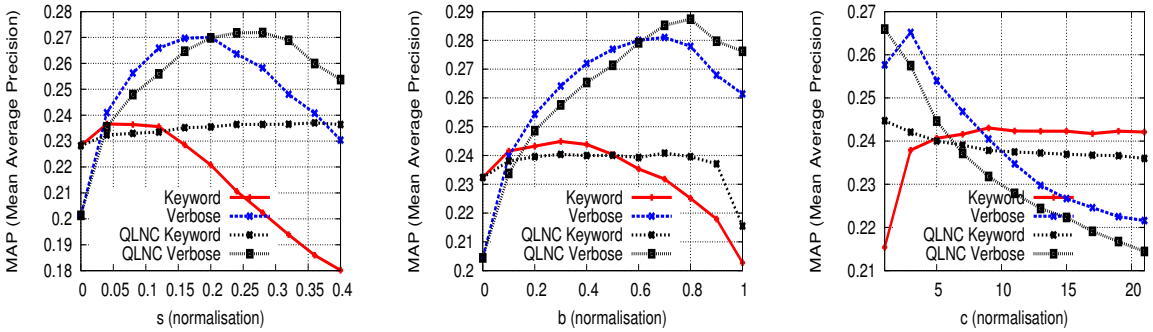

Figure 2: Effectiveness (Mean Average Precision) of Piv, BM25, and PL2 (left to right) vs adapted versions of each Retrieval Function (labelled QLNC) for both Keyword and Verbose queries on FT Collection

component can be interpreted as using both the probability of seeing a specific q in this D (i.e. $|D|/|C|$) and the prior probability of seeing any query term $q \in Q$ in any document D. It is worth remarking that $P(q \in any\_D) \cdot N$ is an accurate estimate of the size of the returned set ($|RET|$) of the query (under the assumption of term-independence). Previous research [8] has suggested that the size of the returned set is correlated to the optimal level of document length normalisation to apply (although in that research a solution to automatically tuning normalisation was not proposed).

4.2 Empirical Evidence of Automatic Tuning

Figures 1 and 2 shows the effectiveness of the adapted retrieval functions compared to the original retrieval functions in terms of MAP (mean average precision) over safe normalisation parameter ranges. In Figures 1 and 2, we can see that the black curves (indicating the performance of the adapted retrieval functions) peak at, or close to, the same parameter values for each specific retrieval function and each specific collection. This is not true for the original versions. For example, for the original BM25 function on the WT2G collection, the best setting for short keyword queries is $b = 0.2$, while for long verbose queries is $b = 0.6$. However, for the adapted BM25 function, the best setting for both long and short queries is $b = 0.6$. The approach adopted has been successful in regulating the level of normalisation so that the same parameter setting is suitable for different queries on the same collection. Furthermore, for the adapted Piv function, we can see that its' effectiveness is less sensitive to the normalisation parameter s over its safe range of values (0 to 0.4 [10]). For all of the adapted retrieval functions, we can see that the optimal level of normalisation to apply for queries of different length is quite similar on the same collection.

5. EXPERIMENTAL RESULTS

In this section, we compare the adapted versions of the retrieval functions against the original functions using a more standard evaluation.

5.1 Test Collections and Baselines

Table 2: Test Collection Characteristics

	WSJ	FBIS	FR	FT	WT2G	WT10G
# Docs	173,253	130,471	55,630	210,158	221,066	1,692,096
avg_dl	206	240	333	190	623	352
$\sigma(dl)$	218	459	508	173	1472	991
Topics	051-200	301-450	251-450	251-450	401-450	451-550
# Topics	149	116	91	188	50	100

The test collections used in this work are subsets of TREC disks 1-5 and two Web collections. We used many collections with varying document length characteristics (Table 2). This aids in drawing more general conclusions. We created three query types. We used short keyword queries (title field only), long keyword queries (description field only),

and verbose queries (title, description, and narrative fields). Porter's stemming and stop-word removal was performed on all collections and queries.

As baseline retrieval functions, we used the suggested default settings for Piv ($s = 0.2$), BM25 ($k_1 = 1.2$, $b = 0.75$, $k_3 = 1000$), and PL2 ($c = 2.0$). We also used tuned versions of the baselines. We tuned s in Piv from 0 to 0.4[2] in increments of 0.04 for each set of queries on each collection (denoted Piv^t). We tuned b in BM25 from 0 to 1 in increments of 0.1 on each collection for each set of queries (denoted BM25^t). We tuned c in PL2 from 1 to 23 in increments of 2.0 on each collection for each set of queries (denoted PL2^t). As the tuning was conducted on each collection and query set, we are confident that the effectiveness of the tuned version of the function is at the upper bound of each respective function. Furthermore, considerable effort is spent tuning these values, which is not afforded to the adapted versions of the retrieval functions that adhere to **QLNC**. The adapted versions of the retrieval functions are denoted Piv^{qn}, BM25^{qn}, and PL2^{qn} respectively, and their normalisation parameters are set to the default values of $s = 0.2$, $b = 0.75$, and $c = 2.0$.

5.2 Comparative Results

Table 3: Effectiveness (MAP) for Keyword Queries

	WSJ	FBIS	FR	FT	WT2G	WT10G
Short Keyword Queries						
Piv	0.2243	0.2166	0.2465	0.2207	0.1916	0.1409
Piv^t	0.2402 †	0.2454 †	0.2746	0.2365 †	0.2633 †	0.1866 †
Piv^{qn}	0.2407 †	0.2544 †	0.2695	0.2354 †	0.2760 †	0.1823 †
BM25	0.2228	0.2306	0.2856	0.2285	0.2397	0.1707
BM25^t	0.2446 †	0.2579 †	0.2875	0.2432 †	0.2963 †	0.1897 †
BM25^{qn}	0.2330 †	0.2633 †	0.2961	0.2406 †	0.2902 †	0.1837 †
PL2	0.2267	0.2240	0.2696	0.2304	0.2350	0.1739
PL2^t	0.2362 †	0.2621 †	0.2954	0.2430 †	0.3018 †	0.1922 †
PL2^{qn}	0.2367 †	0.2653 †	0.2881	0.2426 †	0.3043 †	0.1839 †
Long Keyword Queries						
Piv	0.2011	0.1892	0.2384	0.2158	0.1886	0.1335
Piv^t	0.2087 †	0.2105†	0.2430	0.2173	0.2232 †	0.1640 †
Piv^{qn}	0.2098 †	0.2098	0.2469	0.2139	0.2210 †	0.1599 †
BM25	0.2059	0.2045	0.2620	0.2181	0.2281	0.1550
BM25^t	0.2168	0.2243†	0.2680	0.2253	0.2622 †	0.1723 †
BM25^{qn}	0.2097	0.2247†	0.2804	0.2195	0.2570 †	0.1576
PL2	0.1981	0.1793	0.2191	0.2034	0.2148	0.1460
PL2^t	0.1982	0.1994†	0.2290	0.2038	0.2424 †	0.1485
PL2^{qn}	0.1976	0.1998†	0.2298	0.1952	0.2441 †	0.1462
Verbose Queries						
	WSJ	FBIS	FR	FT	WT2G	WT10G
Piv	0.3113	0.2440	0.3003	0.2700	0.2160	0.1999
Piv^t	0.3135	0.2451	0.3003	0.2700	0.2298 †	0.2172
Piv^{qn}	0.3143	0.2550	0.3056	0.2698	0.2248 †	0.2191 †
BM25	0.3189	0.2575	0.3259	0.2777	0.2701	0.2262
BM25^t	0.3217	0.2592	0.3248	0.2810	0.2824 †	0.2342 †
BM25^{qn}	0.3230	0.2677 †	0.3333	0.2859	0.2785 †	0.2317 †
PL2	0.3067	0.2340	0.3069	0.2733	0.2469	0.2178
PL2^t	0.3101	0.2365	0.3169	0.2651	0.2693 †	0.2210
PL2^{qn}	0.3082	0.2435	0.3172	0.2670	0.2701 †	0.2220

Table 3 shows the effectiveness of the retrieval approaches (Piv, BM25, and PL2) with and without the per-query tuning. We can see that in most cases the adapted versions of the retrieval function significantly[3] outperform the baseline functions.

[2]Previous research has indicated that Piv performs poorly when $s > 0.4$ [10].

[3]† indicates statistical significance at the 0.05 level using a one-tailed t-test compared to the default retrieval function.

6. DISCUSSION AND CONCLUSION

We have introduced a new constraint that formalises the interaction between query-length and document-length normalisation. We have adapted a number of modern retrieval functions so that they automatically adhere to this constraint. Furthermore, we have shown that the adapted retrieval functions perform comparably to tuned versions of the original functions. Although we have introduced a method to automatically tune normalisation on a per-query basis, it may be possible to further improve performance by tuning on a per-collection basis also. This is left for future work.

Acknowledgments

The first author is funded by the Irish Research Council for Science, Engineering and Technology (IRCSET), co-funded by Marie Curie Actions under FP7. The authors would like to thank ChengXiang Zhai and Yuanhua Lv for some informal discussion on query constraints.

7. REFERENCES

[1] Gianni Amati and Cornelis Joost Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Trans. Inf. Syst.*, 20:357–389, October 2002.

[2] Abdur Chowdhury, M. Catherine McCabe, David Grossman, and Ophir Frieder. Document normalization revisited. In *SIGIR '02*, pages 381–382, New York, NY, USA, 2002. ACM.

[3] Tze Leung Chung, Robert Wing Pong Luk, Kam Fai Wong, Kui Lam Kwok, and Dik Lun Lee. Adapting pivoted document-length normalization for query size: Experiments in chinese and english. *ACM Transactions on Asian Language Information Processing*, 5(3):245–263, September 2006.

[4] Stéphane Clinchant and Éric Gaussier. Do ir models satisfy the tdc retrieval constraint. In *SIGIR*, pages 1155–1156, 2011.

[5] Stéphane Clinchant and Éric Gaussier. Retrieval constraints and word frequency distributions a log-logistic model for ir. *Inf. Retr.*, 14(1):5–25, 2011.

[6] Ronan Cummins and Colm O'Riordan. An axiomatic comparison of learned term-weighting schemes in information retrieval: clarifications and extensions. *Artif. Intell. Rev.*, 28(1):51–68, June 2007.

[7] Ronan Cummins and Colm O'Riordan. Measuring constraint violations in information retrieval. In *SIGIR*, pages 722–723, 2009.

[8] Ronan Cummins and Colm O'Riordan. The effect of query length on normalisation in information retrieval. In *AICS'09*, pages 26–32, Berlin, Heidelberg, 2010. Springer-Verlag.

[9] Hui Fang, Tao Tao, and ChengXiang Zhai. A formal study of information retrieval heuristics. In *SIGIR '04*, pages 49–56, New York, NY, USA, 2004. ACM.

[10] Hui Fang and ChengXiang Zhai. An exploration of axiomatic approaches to information retrieval. In *SIGIR*, pages 480–487, 2005.

[11] Ben HE and Iadh Ounis. A study of parameter tuning for term frequency normalization. In *CIKM '03*, pages 10–16, New York, NY, USA, 2003. ACM.

[12] Ben He and Iadh Ounis. Term frequency normalisation tuning for bm25 and dfr models. In *Proceedings of the 27th European conference on Advances in Information Retrieval Research*, ECIR'05, pages 200–214, Berlin, Heidelberg, 2005. Springer-Verlag.

[13] K. Sparck Jones, S. Walker, and S. E. Robertson. A probabilistic model of information retrieval: development and comparative experiments. *Inf. Process. Manage.*, 36(6):779–808, November 2000.

[14] Yuanhua Lv and ChengXiang Zhai. Lower-bounding term frequency normalization. In *CIKM '11*, pages 7–16, New York, NY, USA, 2011. ACM.

[15] Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. In *SIGIR '96*, pages 21–29, New York, NY, USA, 1996. ACM.